How entropy-theorems can show that approximating high-dim Pareto-fronts is too hard

Olivier Teytaud

How entropy-theorems can show that off-line approximating high-dim Pareto-fronts is too hard

Olivier Teytaud

TAO-inria, LRI, UMR 8623(CNRS - Universite Paris-Sud),
bat 490 Universite Paris-Sud 91405 Orsay Cedex France

Abstract. It is empirically established that multiobjective evolutionary algorithms do not scale well with the number of conflicting objectives. We here show that the convergence rate of any comparison-based multi-objective algorithm, for the Hausdorff distance, is not much better than the convergence rate of the random search, unless the number of objectives is very moderate, in a framework in which the stronger assumption is that the objectives have conflicts. Our conclusions are (i) the relevance of the number of conflicting objectives (ii) the relevance of random-search-based criterions (iii) the very-hardness of more than 3-objectives optimization (iv) some hints about new cross-over operators.

1 Introduction

Multi-objective optimization (MOO,[1, 10, 3]) is the research of the set $\{x; \nexists y; (\forall i\ f_i(x) \leq f_i(y)) \wedge (\exists i\ f_i(x) < f_i(y))\}$ which is called the Pareto set. Many papers have been devoted to MOO, some of them with deterministic methods (see [10]), and some others with evolutionary algorithms (EA) ([3]). Usually, Evolutionary MOO (EMOO) is considered as an offline tool for approximating the whole Pareto-sets. Hence, the diversity of the population is a main goal of EMOO ([13]); the goal is a convergence to the whole set. Measuring this convergence to the whole set is difficult as defining quality-criterions is hard ([15]). Convergence proofs and convergence rates exist in non-population-based iterative deterministic algorithms (see e.g. [10, chap.3]), or for specific cases in population-based methods (see e.g. [9]), or very pessimistic-bounds in the case of the discrete domain $\{0, 1\}^n$ ([6]). Empirical results mainly show that scaling up with the number of objectives is not easy ([12],[4]).

The goal of off-line population-based methods is the convergence to the whole Pareto-set, whereas on-line methods lead to iterative procedures in which iteratively (1) the user provides a weighting of the objectives (2) the MOO-algorithm provides an optimum of the corresponding weighted average. We will here investigate conditions under which such a global convergence to the whole Pareto-set is tractable. We will restrict our attention to comparison-based methods, but we conjecture that the comparison-based-nature of the algorithm is indeed not crucial in the results.

We here precisely show (i) an upper bound for a simple random search algorithm (section 2) and (ii) a lower bound for any comparison-based algorithms that is very close to the convergence rate of random search (section 3) when the number of objectives is large. The lower-bound is based on entropy-theorems, a family of results that has been adapted to EA in [11]. The main conclusion is that for our criterion (the Hausdorff distance) EMOO has a strong curse of dimensionality, which is redhibitory for dimension[1] roughly > 3 or 4, except when the problem is such that random search can handle it.

An interesting point is that the "real" number of objectives among a set of d objectives, as the number of dimensions for a subset of $\mathbb{R}^d$, can be studied more carefully than by just bounding it by d. When some objectives are almost equal, then the "true" dimensionality is much lower. In particular, in all our negative results below, we consider that objectives can be conflicting. This is the stronger hypothesis of our work, and our negative results under this hypothesis strongly emphasize the interest of approachs dealing wich non-conflicting hypothesis. Deeply, our work uses packing numbers of pareto-sets; the logarithm of this packing numbers is polynomial with degree the number of conflicting objectives[2].

Related works include (i) papers trying to remove objectives that are not in conflict with others and can therefore be removed without significant change in the Pareto-set [2] (ii) criterions relating the efficiency of a MOO-algorithm to the efficiency of random-search [7].

Notations and definitions

MOO problems are formulated as follows. The (multi-valued) fitness is an application from a given domain to $[0,1]^d$; $d = 1$ is the mono-objective cases, $d > 1$ is a strictly multi-objective problem. We consider that fitnesses, to be maximized, have values in $[0,1]$. A distribution is given, that leads to a distribution of probability P in the space of fitnesses, namely $[0,1]^d$. We note $d(x,y)$ the euclidean

[1] "Dimension" refers to the dimension of the fitness space, i.e. the number of objectives.

[2] By the way, this degree might be non-integer for problems in which some objectives are somewhere between fully conflicting and strictly non-conflicting.

distance between elements $x, y \in [0,1]^d$. We note $d(A,B)$ the Hausdorff-distance between subsets of $\mathbb{R}^d$, i.e.

$$d(A,B) = \max(\sup_{a\in A}\inf_{b\in B} d(a,b), \sup_{b\in B}\inf_{a\in A} d(a,b))$$

(see figure 1, left). We will use the Hausdorff distance in the space of fitnesses.

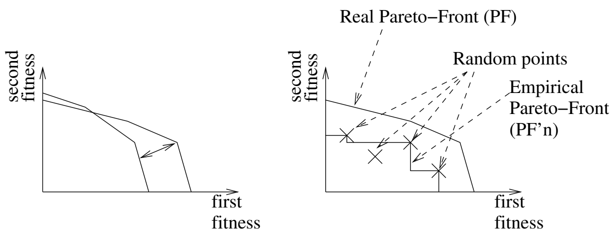

Fig. 1. Left: the Hausdorff distance between two sets. The Hausdorff distance between two Pareto-fronts A and B is the maximal distance between $x \in A$ and B or $x \in B$ and A. In particular, if the distance is ϵ, any ball of radius $> \epsilon$ whose center is in A (resp. B) intersects B (resp. A). Right: illustration of theorem 1.

We note $a \succ b$ if and only if $\forall i, a_i \geq b_i$ and $\exists i, a_i > b_i$. We note $a \succeq b$ if and only if $\forall i, a_i \geq b_i$. If $E \subset [0,1]^d$, we note $PF(E)$ the set of elements dominated by E, i.e. $PF(E) = \{f \in [0,1]^d \text{ s.t. } \exists e \in E, e \succeq f\}$. If P is a distribution, we note $PF = PF(support(P))$ (we omit the index P for short). Being given $e > 0$ and $m(.,.)$ a metric, a e-separated set for $m(.,.)$ is a set S such that $\forall (x,y) \in S, x \neq y \Rightarrow m(x,y) \geq e$. In all the sequel if G is included in a metric space with distance $m(.,.)$ we note $N(G, e, m(.,.))$ the packing number of the set G for $e > 0$ and for metric $m(.,.)$, i.e. the maximal e-separated (for $m(.,.)$) set included in G. If $||.||$ is a norm, then we note $N(G, e, ||.||) = N(G, e, (x,y) \mapsto ||x-y||)$.

We consider a comparison-based EMOO, in the sense that the only use of computed fitness-values is a comparison for the relation $\succeq$. The behavior of the algorithm is therefore only dependent on (i) binary answers to dominance-requests (ii) random choices. P_n is the population proposed by the algorithm as an approximation of the PF. All the laws for randomly generated elements, all the information flow depend on the result of comparisons only. This is an usual (yet not exclusive) framework for EA.

We will study in the sequel the convergence rate of EMOO. This convergence rate is with respect to time. Time is at least linear in the number of tests and in the number of calls to the fitness function. Therefore, we will count as one time step a step which contains either a comparison or a fitness-evaluation (or both). We will show an upper bound (for a naive algorithm) and a (general) lower bound, that are very close to each other when the number of objectives is large. The upper bound is shown on the most simple possible algorithm : the pure random search. We will note $PF_n = PF(fitness(P_n))$ the Pareto-front estimated by the algorithm after n calls to comparison operators.

We consider $\mathcal{P}$ a family of possible problems with d objectives. For each problem:

- a fitness $fitness$ is given from the domain to $[0,1]^d$;
- $PF = \{x \in [0,1]^d; \exists y, y \succeq x\}$.

The precision of the algorithm after n comparisons with confidence $1-\delta$ is the smallest $\epsilon_{\mathcal{P},n,\delta}$ (we will use ϵ for short in the sequel, omitting the dependency in $\mathcal{P}$, n and δ) such that for any $p \in \mathcal{P}$,

$$P(d(PF, PF_n) > \epsilon) \leq \delta$$

2 Upper bounds for the random-search

In this section we (i) define a simple random-search algorithm (ii) evaluate its convergence rate.

An initial empty P_0 is defined. Each epoch for $n = 0, \ldots \infty$ is as follows :

- generate one random point x in the domain;
- set $P_{n+1} = \{x\} \cup P_n$.

(the same Pareto-front PF_n would result from a pruning, i.e. if at the second line we only add x if it is not dominated by any point in P_n and if we remove from P_{n+1} points dominated by x)

An immediate property is that P_n dominates the n randomly drawn elements. We now study, thanks to this simple remark, the convergence of the Hausdorff distance between PF and PF_n.

Theorem 1. *Assume that all fitnesses are upper-bounded by $1-c<1$. For any d, there exists a universal constant K such that with probability at least $1-\delta$, for any absolutely continuous distribution of probability P in the fitness space with density lower bounded in its support by $q>0$, $d(PF_n, PF) \leq K\sqrt[d]{e/q}$, where $e = O(d\log(n) - \log(\delta))/n$.*

Proof :

We note $x_1, \ldots, x_n$ the n randomly drawn points in the fitness-space. We note μ the Lebesgue's measure.

First step, $PF_n \subset PF$. Therefore,

$$d(PF_n, PF) = \sup_{x \notin PF_n} d(x, \mathbb{R}^d \setminus PF) \tag{1}$$

(figure 1 (right) illustrates the previously defined PF_n and PF)

Second step. We now consider $\epsilon < c$ and x such that $x \notin PF_n$ and $d(x, [0,1]^d \setminus PF) > \epsilon$. Consider $x^+ = \{y \in [0,1]^d; y \succeq x\}$. We consider the area $a(x) = \mu(x^+ \cap PF)$. The area $\mu(x^+ \cap B(x,\epsilon))$, where $B(x,\epsilon)$ is $\{z \in [0,1]^d; (d(z,x) < \epsilon)\}$, is $\Omega(\epsilon^d)$. Therefore $a(x) = \Omega(\epsilon^d)$. This concludes the second step.

Third step. We will now use the notion of VC-dimension. Readers unfamiliar with this notion are referred to [5, chap. 12, 13] for an introduction. We will only use VC-dimension to justify equation 2. It is a known fact (see [5, chap. 12,13] that the set $\{a^+ = [a_1, 1] \times [a_2, 1] \times \cdots \times [a_d, 1]; a \in [0,1]^d\}$ has VC-dimension $\leq d$ (see e.g. [5, chap. 13]). This implies that with probability at least $1 - \delta$,

$$\sup_{a \in [0,1]^d; \forall i \in [[1,n]] x_i \notin a^+} P(a^+) \leq e \tag{2}$$

where $e = O(d \log(n) - \log(\delta))/n$.

Fourth step. We now combine previous steps to conclude. Consider $\epsilon = d(PF_n, PF)$. By the first step,

$$\epsilon = \sup_{x \notin PF_n} \inf_{y \notin PF} d(x, y)$$

For any arbitrarily small h, consider some $x \notin PF_n$ realizing this supremum within precision h.

By the third step, as we know that none of the x_i lies in x^+, we know that with probability $1 - \delta$, for any such x, $P(x^+) \leq e = O(d \log(n) - \log(\delta))/n$, and therefore

$$\mu(x^+ \cap PF) = O(d \log(n) - \log(\delta))/qn \tag{3}$$

By the second step,

$$\mu(x^+ \cap PF) = \Omega((\epsilon - h)^d) \tag{4}$$

and therefore at the limit of $h \to 0$, combining equations 3 and 4 leads to $\epsilon^d = O(d \log(n) - \log(\delta))/qn$, hence the expected result. ■

3 Lower bounds for any EA

We will prove lower bounds on the efficiency of comparison-based MOO algorithms and show that these lower bounds are not far from the performance of random search when the number of objectives increase.

We consider families $\mathcal{P}$ of problems as defined in section 1, and a strongly restricted family of problems is enough for concluding: we assume that the fitness space has the same dimensionality as the input space and that the fitness in $\mathcal{P}$ verify the following:

- $fitness(x) = x$ if $x \in PF$;
- $fitness(x) = 0$ otherwise else.

Note that with this particular fitness function, the Pareto-set (in the space of individuals) and the Pareto-fronts (in the space of fitnesses) are equal. The result holds *a fortiori* if we consider a framework in which the domain has higher dimension, or if the relation between x and $fitness(x)$ is more complicated when $fitness(x)$ is in the Pareto-front. Therefore, the negative results are more general than this particular case.

So, we consider a very simple form of MOO problem. We will have no restriction on the EMOO provided that it is comparison-based as defined in section 1. We will consider the number of fitness-comparisons or fitness-evaluations necessary for ensuring with precision at least $1-\delta$ a precision ϵ for the Hausdorff-distance between the output of the algorithm and the target-Pareto-front for any C^1 Pareto-front.

In order to simplify notations (in the case $n = 0$ of the algorithm below), we note $x_{-1} = 0 \in \mathbb{R}^d$. Consider a EMOO, fitting in the following framework:

- initialize s to the empty vector.
- for $n = 0$ to ∞ :
  - generate one individual x_n according to some law $p_n(s)$.
  - update the internal state s by $s \leftarrow (s, \text{`}generate', x_n)$ (we keep in memory the fact that we have generated x_n).
  - compare the fitness of x_n to $fitness(x_{g_n(s)})$ with modality $g'_n(s)$, i.e.:
    - $*$ tests if $fitness(x_n) \succ fitness(x_{g_n(s)})$ (case $g'_n(s) = 0$);
    - $*$ or tests if $fitness(x_{g_n(s)}) \succ fitness(x_n)$ (case $g'_n(s) = 1$)

    where $g_n(s) < n$ and $g'_n(s) \in \{0, 1\}$ and note r the result.
  - update the internal state s by $s \leftarrow (s, \text{`}compare', x, r)$ (we keep in memory the fact that we have compared x to $x_{g_n(s)}$ with modality $g'_n(s)$ and that the result was r.
  - suggest $PF_n = PF(fitness(x_0), \ldots, fitness(x_n))$ as an approximation of the Pareto front.

This covers all multi-objective algorithms based on comparisons only, with various functions $p_n(.)$ and $g_n(.)$. The case of the random search is handled by

$p_n(s)$ constant for any s and n. We can include in this framework any niching mechanism or diversity criterion in the domain. We have not considered different comparisons for $a \succ b$ and $a \succeq b$, but we could consider any set of comparisons provided that the number of possible outcomes is finite; this is just a constant value in the theorem below instead of the 2 in $\log(2)$.

The algorithm has precision ϵ within time n with confidence $1-\delta$ on a given set of problems, if for any problem in some given family of problems, with probability $1-\delta$, $d(PF_n, PF) \leq \epsilon$.

Theorem 2: entropy-theorem for EMOO. *We note $\mathcal{F}$ the set of all $PF(E)$ for any $E \subset [0,1]^d$ such that the Pareto-front is C^1. Assume that for any $s \in \mathcal{F}$, $\mathcal{P}$ contains at least one distribution such that $PF(P) = s$. Then, the number of comparisons required for a precision ϵ and confidence $1-\delta$ is at least $\Omega(1/\epsilon^{d-1}) + \log(1-\delta)/\log(2)$.*

Remark : we could consider a three-outputs-comparison also (one for $a \succ b$, one for $b \succ a$, and one if $a \not\succ b$ and $b \not\succ a$), leading to a factor $\log(2)/\log(3)$ on the bound.

Proof :

Thanks to the lemma below, consider a ϵ-separated set $s_1, \ldots, s_N$ in $\mathcal{F}$ equiped with the Hausdorff-metric, of size $N = \exp(\Omega(1/\epsilon^{d-1}))$.

Consider r the sequence of the n first answers of the algorithm to requests of the form "does $a \succ b$ hold ?". r is a sequence in $\{0,1\}^n$ (r of course depends on the problem and can be random[3]). Note ${PF_n}^r$ the Pareto-front provided by the algorithm if the answers are r. ${PF_n}^r$ is a random variable as the algorithm might be randomized.

First, let's consider a fixed r, in the set R of all possible sequences of answers.

Consider s a random uniform variable in $\{s_1, \ldots, s_N\}$. Consider the probability that ${PF_n}^r$ is at distance $< \epsilon$ of s. This is a probability both on ${PF_n}^r$ and on s. Then,

$$P(d({PF_n}^r, s) < \epsilon) \leq 1/N$$

Now, we will sum on all possible $r \in R$.

$$P(\exists r \in R; d({PF_n}^r, s) < \epsilon) \leq \underbrace{2^n}_{=|\{0,1\}^n|} /N$$

Therefore, this probability can only be $\geq 1-\delta$ if $2^n/N \geq 1-\delta$, therefore $n \log(2) \geq \log(N) + \log(1-\delta)$

[3] As previously pointed out, we could consider a richer comparison-operator with outputs in $\{a \succ b, b \succ a, a \succeq b, b \succeq a, a = b, a \text{ not comparable to } b\}$; this only changes the constant in the theorem.

■

Lemma 1: *The packing number* $N(\mathcal{F}, \epsilon, d(.,.))$ *of the set* $\mathcal{F}$ *with respect to the Hausdorff distance for Lebesgue measure verifies* $\log(N(\epsilon)) = \Omega(1/\epsilon^{d-1})$.

Proof:

Before the proof itself, let's see a sketch of the proof. The packing numbers of derivable spaces of functions are known for the $||.||_\infty$ norm since [8]. The packing numbers of their subgraph are nearly the same thanks to a lemma below. The proof will then be complete. Now, let's go to the details.

Consider the set F of applications $f : [0,1]^{d-1} \to [0,1]$ which are C^1 with (i) derivative with respect to any coordinate bounded by $1/(4d^2)$ (ii) value bounded by $1/(4d^2)$.

For any fixed f, consider $p_f = \{(x,y) \in [0,1]^{d-1} \times [0,1]; y \leq g(x)\}$, where $g(x) = \frac{1}{2} + f(x) - \frac{1}{2d^2}\sum_i x_i$. As $g(x) \in [0,1]$ and $\frac{\partial g(x)}{\partial x_i} < 0$, we see that $p_f \in \mathcal{F}$.

The proof is now the consequence of (i) the lemma below relating the packing numbers of the C^1-functions in F and the packing numbers of their subgraphs $\{p_f; f \in F\} \subset \mathcal{F}$ for the Hausdorff-metric (ii) the bound $N(F, \epsilon, ||.||_\infty) = \Omega(1/\epsilon^{d-1})$ provided in [8] (see also [14, 5] for more recent references). ■

Lemma 2: *Consider a fixed* d*. Then for some constant* C*,*

$$N(\{p_f; f \in F\}, \epsilon, d(.,.)) \geq C \times N(F, \epsilon, ||.||_\infty)$$

Proof: All we need is

$$d(p_{f_1}, p_{f_2}) = \Omega(||f_1 - f_2||_\infty)$$

for functions in F. The sequel of the proof is devoted to proving this inequality. The proof is as follows :

1. let $\delta = ||f_1 - f_2||_\infty$.
2. by compactness, δ is realized by some $x : |f_1(x) - f_2(x)| = \delta$. Without loss of generality, we can assume $f_1(x) = f_2(x) + \delta$.
3. consider $g_i : t \mapsto \frac{1}{2} + f_i(t) - \frac{1}{2d^2}$. As pointed out in the proof of the lemma , the subgraph of g_i is p_{f_i} (by definition).
4. then $g_1(x) - g_2(x) = \delta$.
5. consider the euclidean distance δ_2 between $(x, g_1(x))$ (which is in p_{f_1}) and p_{f_2}.
6. this distance is realized (thanks to compactness) by some $z : \delta_2 = d((z, g_2(z)), (x, g_1(x)))$.
7. by the bound on the derivatives of the g_i (which have the same derivatives as the f_i) and by step 2, $g_1(x) - g_2(z) \geq \delta - K(d(z,x))$ for some K.

8. then, $\delta_2^2 = d(z,x)^2 + (g_1(x) - g_2(z))^2 \geq \max(d(z,x)^2, \max(0, \delta - Kd(z,x))^2)$.
9. there are now two cases:
   - $d(z,x) < \delta/(2K)$, and then $\delta - Kd(z,x) \geq \delta/2$ and ${\delta_2}^2 \geq \delta^2/4$ (by step 7).
   - $d(z,x) \geq \delta/(2K)$, and then $\delta_2^2 \geq d(z,x)^2 \geq \delta^2/(4K^2)$ by step 8.

   and this implies in both cases that $\delta_2 \geq \min(\delta/2, \delta/(2K)) = \Omega(\delta)$.

The proof is complete. ■

4 Conclusion

We have shown a lower bound on the complexity of finding a Pareto-set within precision ϵ for the Hausdorff-distance, that holds for any comparison-based algorithms, and that almost matches the complexity of random search when d is large. Let's examine precisely the results depending on the dimension. Note N_R the number of comparisons required for the random search, and N_E the number of pairs (comparisons,fitness-evaluations) required for the EA. Compare these two numbers for a given precision ϵ going to 0. Then, $N_E \geq N_R^{\frac{d-1}{d}}$. For $d = 1$, this allows the well-known difference between the slow convergence of random search and established linear convergence rates for mono-objective EA. For $d = 2$, this is still satisfactory: $N_E \geq N_R^{\frac{1}{2}}$. We can be much faster than the random search. For $d = 10$, this leads to $N_E \geq N_R^{\frac{9}{10}}$.

This disappointing result has to be discussed. The random-search algorithm as previously defined, provides a good solution in terms of the Hausdorff distance at least if the number of generations is sufficient, but this solution is far from parsimonious. It contains many elements, only a small part of them being non-dominated by others. So, we compare only solutions provided by random-search and any comparison-based algorithms in a framework in which parsimony is not required. We compare the computation time before the algorithms provide a description of a not-too-bad Pareto-front for the Hausdorff distance, without regarding the size of the description of the Pareto-front. The random search provides a non-compact description. This is the strongest limitation of this work. This implies that the main result of this work is that in dimension d large, the rule used for selecting new candidates is not much better than the pure random search. This does not imply, of course, that various techniques are not helpful, but mainly these techniques will prune the solution efficiently, and not significantly improve the convergence rate in terms of Hausdorf distance with respect to the sum number of comparisons plus number of fitness-evaluations (this sum lower-bounds the computation-time).

A strength of the results in this paper has to be emphasized. We have considered entropy of smooth Pareto-fronts and it was enough to deduce strong lower bounds. The entropy numbers are the same if we add various constraints, e.g. convexity or concavity of the Pareto-front. Therefore, we can not get an improvement without strong hypothesis, different from convexity or smoothness. It is necessary, to get rid of the limits proved in this paper, to restrict one's attention to spaces of Pareto-fronts with small entropy numbers (typically smoothly parametric Pareto-fronts).

The following question naturally arises: can our results be extended to non-comparison-based algorithms ? We believe that gradient-based informations on the hyperplane tangent to the Pareto-front can lead to theoretically interesting improvements, in which the solutions are no more described by a population of points close to the frontier but by local approximations based on a population of points. This requires however (i) the use of fitness-values (and not only comparisons) or (ii) the use of gradient information, that would probably be not very reliable for real-world applications. We therefore believe that generalizations of our results are possible, that would show that unless very strong smoothness hypothesis, MOO in large dimension requires interactive sessions in which the user guides the research in interesting areas of the Pareto-front. Deeply, it is likely that there's no description of Pareto-fronts that would be useful in large dimension and that can fit in memory or even be written in memory in tractable time.

We point out the following elements, summarizing the elements above and adding some other ones:

- our results do not take care about compactness of the solution provided by algorithms; they only consider the time that is required before a good solution is found, in terms of the Hausdorff metric, but not the time that is required before a good and parsimonious solution is found;
- we conjecture that our results are not specific of comparison-based methods; however, we could not extend the results to a more general case;
- we consider convergence in terms of the Hausdorff distance to the whole Pareto-set; this is not related to other forms of MOO like interactive-MOO.
- the computation-time is lower bounded by the number of comparisons; this assumption of our theorems of course hold, but when almost all the computation time is indeed in the computation of the fitnesses, then this might be a bad model. Therefore, for expensive optimization (typically, when 3 hours are required for computing the fitnesses of an individual), our results might be misleading.

Assuming that the conjecture (line 2 above) holds (unless we should precise "in comparison-based methods", our results could therefore be applied as follows:

when the number of conflicting objectives is high, online interactions with the user, in order to specify the interesting part of the Pareto-set, are necessary.

Finally, our result is a form of No-Free-Lunch theorem for MOO with some realistic distribution of conflicting fitnesses. We do not show that all algorithms are equivalent, but we show that all comparison-based algorithms have a similar behavior (the same order of magnitude for computation-time) when the dimension increases. This has three main applications in practice.

- The first one is that the real dimensionality of a multi-objective problem is the number of conflicting objectives. This is strongly related to works in the direction of the removal of non-conflicting objectives[2].
- A second point is the relevance of criterions relating the efficiency of a MOO-algorithm to the efficiency of random-search [7]. Whereas for mono-objective algorithms, there are perhaps too much orders of magnitudes between random search and efficient algorithms, in the MOO-case such measure is reasonnable. Our work shows that for a large number of objectives, if the improvement on random search is huge, then the algorithm uses some regularity of the problem - as pointed out above, mainly the number of non-conflicting objectives.
- The third more subtle point concerns the comparison operator. Our work deals with binary comparisons, but in the multi-objective case more subtle comparison operator, comparing each fitness separately, could be considered. Instead of two bits for the cases $a \succ b$, $b \succ a$, a and b not comparable, one could consider d bits of information, each bit being the comparison between a_i and b_i. However, results in the paper are mainly preserved; the lower-bounded on the computation time is at most divided by a factor d when such "stronger" comparison operators are applied. But, for e.g. 4 or 5 conflicting objectives, we do believe that such improvements are possible by the definition of cross-over operators that use this additional information, as well as some constraint-handling techniques use the full constraint-violation information and not only one bit for the satisfaction of all constraints. As far as we know, no such multi-objective cross-over has been proposed yet.

Acknowledgements

This work was supported in part by the Pascal Network of Excellence, and benefited from fruitful discussions with Marc Schoenauer and from constructive and helpful remarks from anonymous reviewers.

References

1. S. Bhattacharyya. Evolutionary algorithms in data mining: multi-objective performance modeling for direct marketing. In *Proceedings of KDD 2000, pp 465-473*, 2000.
2. Dimo Brockhoff and Eckart Zitzler. Are all objectives necessary? on dimensionality reduction in evolutionary multiobjective optimization. In *proceedings of PPSN*, 2006.
3. K. Deb. *Multi-Objective Optimization Using Evolutionary Algorithms*. John Wiley, 2001.
4. K. Deb, A. Sinha, and S. Kukkonen. Multi-objective test problems, linkages, and evolutionary methodologies. Technical report, KanGAL Report No. 2006001, 2006.
5. L. Devroye, L. Györfi, and G. Lugosi. *A probabilistic Theory of Pattern Recognition*. Springer, 1997.
6. Oliver Giel. Runtime analysis of a simple multi-objective evolutionary algorithm. In Jürgen Branke, Kalyanmoy Deb, Kaisa Miettinen, and Ralph E. Steuer, editors, *Practical Approaches to Multi-Objective Optimization*, number 04461 in Dagstuhl Seminar Proceedings. Internationales Begegnungs- und Forschungszentrum (IBFI), Schloss Dagstuhl, Germany, 2005. <http://drops.dagstuhl.de/opus/volltexte/2005/271> [date of citation: 2005-01-01].
7. Evan J. Hughes. Multi-objective equivalent random search. In *proceedings of PPSN*, 2006.
8. A.-N. Kolmogorov and V.-M. Tikhomirov. ϵ-entropy and ϵ-capacity of sets in functional spaces. *Amer. Math. Soc. Transl. 17, pp 277-364*, 1961.
9. M. Laumanns, L. Thiele, E. Zitzler, E. Welzl, and K. Deb. Running time analysis of multi-objective evolutionary algorithms on a simple discrete optimization problem, 2002.
10. K.M. Miettinen. *Nonlinear Multiobjective Optimization*. Kluwer Academic Publishers, Boston, Massachusetts, 1999.
11. S. Gelly O. Teytaud. General lower bounds for evolutionary algorithms. In 10^{th} *International Conference on Parallel Problem Solving from Nature (PPSN 2006)*, 2006.
12. K. Sastry, M. Pelikan, and D.E. Goldberg. Limits of scalability of multiobjective estimation of distribution algorithms. *Proceedings of the Congress on Evolutionary Computation, 3, 2217-2224*, 2005.

13. A. Toffolo and E. Benini. Genetic diversity as an objective in multi-objective evolutionary algorithms. In *Evolutionary Computation, 11(2):151–168*, 2003.
14. A.W. van der Vaart and J.A. Wellner. *Weak Convergence and Empirical Processes, With Applications to Statistics.* Springer Series in Statistics. Springer-Verlag New York, Inc., 1996.
15. E. Zitzler, M. Laumanns, L. Thiele, C. Fonseca, and V. da. Why quality assessment of multiobjective optimizers is difficult, 2002.